Интерполяция. Построение интерполяционного многочлена.

Сразу процитирую Вики: *"Интерполяция, интерполирование (от лат. inter–polis — «разглаженный, подновлённый, обновлённый; преобразованный») — в вычислительной математике способ нахождения промежуточных значений величины по имеющемуся дискретному набору известных значений"*. [https://ru.wikipedia.org/wiki/Интерполяция]

Расшифруем немножко, что же именно там написано. Имеются результаты нескольких измерений некоторой величины при различных значениях некоторого параметра, от которого она зависит. Иными словами, известны значения некоторой функции (вообще говоря, нам неизвестной) при нескольких различных значениях её аргумента. А мы хотим получить, хотя бы приблизительно, значения функции в каких-то промежуточных точках, при значениях аргумента, в которых мы её не измеряли. Понятно, что, формально говоря, значения могут быть любыми, но по жизни так не бывает, функция, описывая некоторое реальное явление или процесс, обычно ведёт себя более или менее плавно, именно разглаженно. Вспомним, как мы начинали знакомиться с функциями, точнее, как нам объясняли, что такое график функции: есть функция (да, заданная формулой, но сейчас это неважно), вычисляем её значения при нескольких значениях аргумента (ну, или в более общем случае, как-то их находим, измеряем, например), потом наносим на координатную плоскость соответствующие точки, а потом соединяем эти точки плавной линией. Ключевое слово здесь *"плавной".* Хотя, конечно, плавность - штука в определённой степени субъективная, но она есть. И уже по построенной линии мы можем приблизительно считывать значения функции в каких-то промежуточных точках.

А что делать, если нам надо таких графиков миллион? Не промежуточных точек на одном графике, а именно графиков, т.е. если у нас есть миллион наборов данных, полученных неважно как – то ли в результате измерений, то ли в результате вычислений. Надо что-то придумывать. И придумка лежит на поверхности - построить какую-то более или менее "нормальную" функцию, таки обладающую достаточно плавным графиком, описать её каким-то алгоритмом вычисления, обычно формулой, причём функция должна соответствовать результатам наших измерений, а потом просто вычислять значения этой функции при необходимых нам значениях аргумента.

Общий подход понятен, но вопросов всё равно остаётся много: какого вида выбрать функцию, как её искать, насколько она адекватна реальной ситуации и т.д. Ограничимся одним важным случаем - важным потому, что он довольно прост и в то же время эффективен, и часто даёт достаточно адекватные результаты - в качестве функции, соответствующей нашим измерениям, выбираем многочлен. Функцию, соответствующую измерениям, график которой проходит через изображённые на координатной плоскости точки, называют по-простому интерполирующей функцией, а сам процесс называют, как мы уже видели, интерполяцией.

Введём обозначения. Обозначим значения аргумента, в которых задана функция, черех x_0, x_1, x_2, x_3, … x_N. Да, точек всего N+1 штук. Почему N+1? А потому, что мы будем интерполировать N+1 точку многочленом N-ой степени, точнее степени, не

выше (не больше) N. Оказывается, и мы это ещё увидим, что такой многочлен всегда существует и притом единственный.

Имеется N+1 значений аргумента (иксов, говоря по-простому) x_0, x_1, x_2, x_3, … x_N, и заданы значения функции (игреков) в этих точках y_0, y_1, y_2, y_3, … y_N: $f(x_0) = y_0$, $f(x_1) = y_1$, $f(x_2) = y_2$, ..., $f(x_N) = y_N$. Среди заданных значений аргумента нет повторяющихся.

Требуется построить многочлен $p(x) = a_0 + a_1x + a_2x^2 + … + a_Nx^N$ степени не выше N, удовлетворяющий соотношениям $p(x_0) = y_0$, $p(x_1) = y_1$, $p(x_2) = y_2$, ..., $p(x_N) = y_N$.

Числовой пример. Описание.

Ниже мы рассмотрим три способа построения интерполяционного многочлена. Все три метода будем демонстрировать и проверять на одном числовом примере. Опишем его здесь.

Мы будем строить многочлен третьей степени, график которого проходит через точки (-2, 1), (1, -4), (2, 5), (4, 3), т.е.

$x_0=-2$, $y_0=1$, $x_1=1$, $y_1=-4$, $x_2=2$, $y_2=5$, $x_3=4$, $y_3=3$.

Метод 0. Сведение задачи к решению системы линейных уравнений.

Необходимое предупреждение. Сразу хочу остановиться, и остановиться очень тщательно, на вопросе обозначений. Обозначения выглядят довольно естественно, никакого внутреннего отторжения не вызывают. И они действительно естественны. Но! При этом получается некоторая (чисто психологическая, этакая ловушка мышления) сложность. Построить многочлен – это значит найти значения коэффициентов a_0, a_1, a_2, … a_N. Получается, что неизвестные у нас - a_0, a_1, a_2, … a_N, а заданные параметры – это x_0, x_1, x_2, … x_N и y_0, y_1, y_2, … y_N. И это действительно так. Добавляет ветра, сносящего крышу, то, что просто x без индекса – это переменная, это обозначение аргумента многочлена p. И очень важно максимально точно, чётко, отчётливо понимать, что есть что в нашей задаче.

А теперь давайте просто распишем условия, которым должен удовлетворять многчлен p.

$$p(x_0) = a_0 + a_1 \cdot x_0 + a_2 \cdot x_0^2 + a_3 \cdot x_0^3 + \ldots + a_N \cdot x_0^N$$
$$p(x_1) = a_0 + a_1 \cdot x_1 + a_2 \cdot x_1^2 + a_3 \cdot x_1^3 + \ldots + a_N \cdot x_1^N$$
$$p(x_2) = a_0 + a_1 \cdot x_2 + a_2 \cdot x_2^2 + a_3 \cdot x_2^3 + \ldots + a_N \cdot x_2^N$$
$$\ldots$$
$$p(x_N) = a_0 + a_1 \cdot x_N + a_2 \cdot x_N^2 + a_3 \cdot x_N^3 + \ldots + a_N \cdot x_N^N$$

Многочлен p должен удовлетворять условиям:

$$p(x_0) = y_0,\ p(x_1) = y_1,\ p(x_2) = y_2,\ \ldots,\ p(x_N) = y_N$$

Это даёт нам систему линейных уравнений:

$$\begin{cases} a_0 + x_0 \cdot a_1 + x_0^2 \cdot a_2 + x_0^3 \cdot a_3 + \ldots + x_0^N \cdot a_N = y_0 \\ a_0 + x_1 \cdot a_1 + x_1^2 \cdot a_2 + x_1^3 \cdot a_3 + \ldots + x_1^N \cdot a_N = y_1 \\ a_0 + x_2 \cdot a_1 + x_2^2 \cdot a_2 + x_2^3 \cdot a_3 + \ldots + x_2^N \cdot a_N = y_2 \\ \qquad\qquad \cdot \cdot \cdot \\ a_0 + x_N \cdot a_1 + x_N^2 \cdot a_2 + x_N^3 \cdot a_3 + \ldots + x_N^N \cdot a_N = y_N \end{cases}$$

Именно линейных! Ничего сложного в этой системе нет, но очень важно здесь остановиться и поговорить о том, что написано в предупреждении в самом начале этого раздела. Я ведь не зря переставил местами a и x при переходе от выражений p(x) в точках x_0, x_1, x_2, … x_N к систему уравнений которую нам надо решить. Так вот, в этой системе уравнений неизвестные, повторю, - это a_0, a_1, a_2, … a_N, а вот как раз x_0, x_1, x_2, … x_N во всяких степенях – это коэффициенты системы, параметры задачи. А y_0, y_1, y_2, … y_N – это правые части уравнений системы, т.е. тоже параметры системы. В сасмом деле, величины x_0, x_1, x_2, … x_N и y_0, y_1, y_2, … y_N нам даны, это входные данные задачи, это точки, в которых заданы значения интерполируемой функции, ну, или точки, которые мы интерполируем. Т.е. в нашей задаче – это её параметры. А вот как раз a_0, a_1, a_2, … a_N - это коэффициенты многочлена, это то, что нам надо найти, это неизвестные в задаче интерполирования. И это момент должен прозвучать явно, очень чётко, и его надо очень отчётливо осознать вот прямо сейчас. Слово “помнить” здесь, на мой взгляд совершенно неуместно, это надо не помнить, типа, “вот так вот обозначили”, это надо именно осознать: иксы и игреки – результаты измерений - данные, a – коэффициенты многочлена – их мы ищем.

Перепишем полученную систему так, как мы это делали в прошлый раз – записывая только характеристики системы - коэффициенты при неизвестных и правые части:

$$\left.\begin{array}{ccccccc} 1 & x_0 & x_0^2 & x_0^3 & \ldots & x_0^N & \\ 1 & x_1 & x_1^2 & x_1^3 & \ldots & x_1^N & \\ 1 & x_2 & x_2^2 & x_2^3 & \ldots & x_2^N & \\ & & & \cdot\ \cdot\ \cdot & & & \\ 1 & x_N & x_N^2 & x_N^3 & \ldots & x_N^N & \end{array}\right| \begin{array}{c} y_0 \\ y_1 \\ y_2 \\ \\ y_N \end{array}$$

И всё. Остаётся только решить полученную систему линейных алгебраических уравнений, получив тем самым коэффициенты интерполяционного многочлена.

Чего же нам ещё? Чем этот метод нехорош? Ответ ожидаемый – есть и получше. Чем, чем получше-то, что здесь может быть получше? Скорость, разумеется. Есть способы построить интерполяционный многочлен существенно быстрее.
Мелкий шрифт: сложность решения системы линейных уравнений методом Гаусса $O(N^3)$, а мы ниже приведём два способа построения интерполяционного многочлена, оба имеют квадратическую скорость выполнения.

Заметим, возможно, что и без формальнонго доказательства, что для любого набора данных x_0, x_1, x_2, … x_N и y_0, y_1, y_2, … y_N (конечно, при условии что среди иксов нет одинаковых – было бы крайне странно, если бы было не так, учитывая их смысл) существует и притом единственный интерполяционный многочлен, степень которого не превосходит N. Существование такого многочлена мы докажем ниже вполне себе конструктивно – мы просто предложим два метода, строящих этот многочлен. А вот доказательство единственности, хоть оно и очень простое, но его я тоже вынесу в мелкий шрифт - боюсь, что дети не знают, что любой многочлен степени N (N>0) имеет не более N действительных корней (ну, или ровно N комплексных корней с учётом их кратности).
Мелкий шрифт – доказательство единственности интеполяционного многочлена, степени не выше N: допустим мы построили два многочлена p и q, интерполирующих набор точек (x_0, y_0), (x_1, y_1), (x_2, y_2), … (x_N, y_N). Тогда $p(x_0) = q(x_0) = y_0$, $p(x_1) = q(x_1) = y_1$, $p(x_2) = q(x_2) = y_2$, ..., $p(x_N) = q(x_N) = y_N$. Рассмотрим многочлен $w(x) = p(x) – q(x)$. Степень многочлена w, ясное дело, тоже не превосходит N. Кроме того, $w(x_0) = w(x_1) = w(x_2) = ... w(x_N) = 0$, поскольку значения p и q во всех этих точках совпадают. Получили многочлен w, степени не выше N, с N+1 корнем. Это и означает, что w – это нулевой многочлен, т.е. многочлены p и q обязательно совпадают.

Числовой пример. Построение интерполяционного многочлена с помощью решения системы линейных уравнений.

Имеем на входе $x_0=-2$, $y_0=1$, $x_1=1$, $y_1=-4$, $x_2=2$, $y_2=5$, $x_3=4$, $y_3=3$.
Система линейных уравнений получается такой:

1	-2	4	-8	1
1	1	1	1	-4
1	2	4	8	5
1	4	16	64	3

Решим её методом Гаусса. Не будем заморачиваться на поиск главного элемента – вычислительной погрешности у нас не будет, поскольку мы будем считать всё точно, руками, а деления на 0 не возникнет. Ну, так получилось ☺ (На самом деле не возникнет потому, что и не может возникнуть, но не будем вдаваться здесь в эти подробности).

Я не буду приводить числовые выкладки, просто выпишу последовательные таблицы, которые получаются при выполнении прямого хода метода Гаусса:

1	-2	4	-8	1
0	3	-3	9	-5
0	4	0	16	4
0	6	12	72	2

1	-2	4	-8	1
0	3	-3	9	-5
0	0	4	4	10 2/3
0	0	18	54	12

1	-2	4	-8	1
0	3	-3	9	-5
0	0	4	4	10 2/3
0	0	0	36	-36

Распишем обратный ход:
$a_3 = -36/36 = -1$,
$a_2 = (10\ 2/3 - 4\cdot(-1))/4 = 11/3$,
$a_1 = (-5- 9\cdot(-1)-(-3)\cdot 11/3)/3 = 5$,
$a_0 = (1 - (-8)\cdot(-1) - 4\cdot 11/3 - (-2)\cdot 5) = -35/3$.

Итого, получили интерполяционный многочлен $p(x) = -x^3 + 11/3\ x^2 + 5\ x - 35/3$. Можно проверить, действительно $p(-2) = 1$, $p(1) = -4$, $p(2) = 5$, $p(4) = 3$.

Метод 1. Интерполяционный многочлен Лагранжа.

Часто используется именно такое название, хотя точнее называть это построение (и так его тоже называют) интерполяционным многочленом в форме Лагранжа. Понятное дело, что, поскольку многочлен N-ой степени, интерполирующий N+1 точку, единственен, то как его ни строй, получаем одно и тоже. Так что интерполяционный многочлен в форме Лагранжа – это только алгоритм построения и способ записи того самого интерполяционного многочлена. Так что сейчас мы излагаем алгоритм Лагранжа построения интерполяционного многочлена (хотя так его обычно не называют).

Построим следующую систему из N+1 многочлена:

$$p_k(x) = \frac{(x-x_0)\cdot(x-x_1)\cdot\ldots\cdot(x-x_{k-1})\cdot(x-x_{k+1})\cdot\ldots\cdot(x-x_{N-1})\cdot(x-x_N)}{(x_k-x_0)\cdot(x_k-x_1)\cdot\ldots\cdot(x_k-x_{k-1})\cdot(x_k-x_{k+1})\cdot\ldots\cdot(x_k-x_{N-1})\cdot(x_k-x_N)}$$

для k = 0, 1, 2, …N

Чтобы было немножко понятнее, выпишу эти многочлены для конкретных k:

$$p_0(x) = \frac{(x-x_1)\cdot(x-x_2)\cdot\ldots\cdot(x-x_{N-1})\cdot(x-x_N)}{(x_0-x_1)\cdot(x_0-x_2)\cdot\ldots\cdot(x_0-x_{N-1})\cdot(x_0-x_N)}$$

$$p_1(x) = \frac{(x-x_0)\cdot(x-x_2)\cdot(x-x_3)\cdot\ldots\cdot(x-x_{N-1})\cdot(x-x_N)}{(x_1-x_0)\cdot(x_1-x_2)\cdot(x_1-x_3)\cdot\ldots\cdot(x_1-x_{N-1})\cdot(x_1-x_N)}$$

$$p_2(x) = \frac{(x-x_0)\cdot(x-x_1)\cdot(x-x_3)\cdot(x-x_4)\cdot\ldots\cdot(x-x_{N-1})\cdot(x-x_N)}{(x_2-x_0)\cdot(x_2-x_1)\cdot(x_2-x_3)\cdot(x_2-x_4)\cdot\ldots\cdot(x_2-x_{N-1})\cdot(x_2-x_N)}$$

. . .

$$p_N(x) = \frac{(x-x_0)\cdot(x-x_1)\cdot(x-x_2)\cdot\ldots\cdot(x-x_{N-1})}{(x_N-x_0)\cdot(x_N-x_1)\cdot(x_N-x_2)\cdot\ldots\cdot(x_N-x_{N-1})}$$

В принципе, уже сейчас можно начинать решать числовой пример – в этом особенность данного метода построения интерполяционного многочлена: все вот эти вот базисные многочлены можно начинать выписывать ещё до поступления сведений об измерениях – значениях игреков, эти многочлены полностью определяются только значениями аргументов интерполируемой функции (иксов точек интерполяции). Но пока отложим эти вычисления.

Чем замечательны базисные многочлены? Их замечательные свойства видны уже из построения: каждый из них равен 0 во всех точках интерполяции (при всех иксах из набора x_0, x_1, x_2, … x_N), кроме одной единственной точки, в которой он равен 1. При этом k-й многочлен равен 1 в точке x_k. Да, именно так

$$p_k(x_i) = \begin{cases} 0, & i \neq k \\ 1, & i = k \end{cases}$$

Это очень легко проверить, именно с такой целью их и строили, это отчётливо видно из самой конструкции. Я даже писать тут ничего не стану, тут надо просто внимательно посмотреть на запись многочлена p_k – все движущие рычаги построения там видны совершенно отчётливо, так что просто сделайте это - посмотрите внимательно.

И тогда мгновенно получается рецепт построения интерполяцинного многочлена p(x):

$$p(x) = y_0 \cdot p_0(x) + y_1 \cdot p_1(x) + y_2 \cdot p_2(x) + \ldots + y_N \cdot p_N(x)$$

И это всё!

Числовой пример. Построение интерполяционного многочлена Лагранжа.

Имеем на входе x_0=-2, x_1=1, x_2=2, x_3=4. Я умышленно не написал игреки – они нам пока не нужны. Строим многочлены p_0, p_1, p_2 и p_3:

$$p_0(x) = \frac{(x-1)\cdot(x-2)\cdot(x-4)}{(-2-1)\cdot(-2-2)\cdot(-2-4)} = \frac{x^3-7x^2+14x-8}{-72} = -\frac{1}{72}x^3 + \frac{7}{72}x^2 - \frac{7}{36}x + \frac{1}{9}$$

$$p_1(x) = \frac{(x-(-2))\cdot(x-2)\cdot(x-4)}{(1-(-2))\cdot(1-2)\cdot(1-4)} = \frac{x^3-4x^2-4x+16}{9} = \frac{1}{9}x^3 - \frac{4}{9}x^2 - \frac{4}{9}x + \frac{16}{9}$$

$$p_2(x) = \frac{(x-(-2))\cdot(x-1)\cdot(x-4)}{(2-(-2))\cdot(2-1)\cdot(2-4)} = \frac{x^3-3x^2-6x+8}{-8} = -\frac{1}{8}x^3 + \frac{3}{8}x^2 + \frac{3}{4}x - 1$$

$$p_3(x) = \frac{(x-(-2))\cdot(x-1)\cdot(x-2)}{(4-(-2))\cdot(4-1)\cdot(4-2)} = \frac{x^3-x^2-4x+4}{36} = \frac{1}{36}x^3 - \frac{1}{36}x^2 - \frac{1}{9}x + \frac{1}{9}$$

И теперь вспоминаем, что у нас были игреки: y_0=1, y_1=-4, y_2=5, y_3=3.

$$p(x) = p_0(x) - 4p_1(x) + 5p_2(x) + 3p_3(x) =$$

$$= -\frac{1}{72}x^3 + \frac{7}{72}x^2 - \frac{7}{36}x + \frac{1}{9} - 4\left(\frac{1}{9}x^3 - \frac{4}{9}x^2 - \frac{4}{9}x + \frac{16}{9}\right) + 5\left(-\frac{1}{8}x^3 + \frac{3}{8}x^2 + \frac{3}{4}x - 1\right) + 3\left(\frac{1}{36}x^3 - \frac{1}{36}x^2 - \frac{1}{9}x + \frac{1}{9}\right)$$

Собственно, это и есть выражение интерполяционного многочлена в форме Лагранжа.

Но давайте раскроем скобки и приведём полученный многочлен к стандартному виду. Исклчительно для того, чтобы убедиться, что мы получили тот же самый многочлен, что и при решении системы линейных уравнений.

$$p(x) = \left(-\frac{1}{72} - \frac{4}{9} - \frac{5}{8} + \frac{3}{36}\right)x^3 + \left(\frac{7}{72} + \frac{16}{9} + \frac{15}{8} - \frac{3}{36}\right)x^2 + \left(-\frac{7}{36} + \frac{16}{9} + \frac{15}{4} - \frac{3}{9}\right)x + \left(\frac{1}{9} - \frac{64}{9} - 5 + \frac{3}{9}\right) =$$

$$= -x^3 + \frac{11}{3}x^2 + 5x - \frac{35}{3}$$

Убедились.

Метод 2. Интерполяционный многочлен Ньютона.

Повторю, точнее называть эту конструкцию алгоритмом построения интерполяционного многочлена в форме Ньютона.

Здесь идея совсем другая. Мы получаем многочлен N-й степени, интерполирующий N+1 точку постепенно: сначла строим многочлен 0-й степени (это просто константа) – обозначим его q_0, - интерполирующий одну точку (x_0, y_0), затем строим многочлен первой степени q_1, интерполирующий две точки - (x_0, y_0) и (x_1, y_1), потом – многочлен второй степени q_2, интерполирующий точки (x_0, y_0), (x_1, y_1), (x_2, y_2) и т.д. q_N – это и есть интерполяционный многочлен. Фокус тут, разумеется, в том, как переходить на следующий уровень.

Начнём.
Построение q_0 очевидно: должно выполняться условие $q_0(x_0) = y_0$. q_0 – многочлен нулевой степени, константа, так что $q_0(x) = y_0$. На всякий случай, ещё раз обращаю внимание на разницу в записях $q_0(x_0) = y_0$ и $q_0(x) = y_0$. В первом случае x_0 – это некоторое число, параметр задачи, и запись $q_0(x_0) = y_0$ означает, что значение многочлена q_0 в точке x_0 (должно быть) равно y_0; во втором случае x – это обозначение переменной, и запись $q_0(x) = y_0$ – это формула для функции q_0.

А вот q_1 будем искать в виде

$$q_1(x) = q_0(x) + c_1 \cdot (x - x_0)$$

где c_1 – некоторая, пока не известная нам, величина.

Понятно, что $q_1(x_0) = q_0(x_0) + c_1 \cdot (x_0 - x_0) = q_0(x_0) = y_0$
Но нам надо, чтобы ещё выполнялось условие $q_1(x_1) = y_1$. Вот и добьёмся его выполнения за счёт выбора множителя c_1. Имеем $q_1(x_1) = q_0(x_1) + c \cdot (x_1 - x_0) = y_1$. Отсюда

$$c_1 = \frac{y_1 - q_0(x_1)}{x_1 - x_0}$$

Заметим, что $x_1 \neq x_0$ и, следовательно, $x_1 - x_0 \neq 0$.

Теперь ищем q_2 в виде

$$q_2(x) = q_1(x) + c_2 \cdot (x - x_0) \cdot (x - x_1)$$

где множитель c_2 нам пока не известен.
Очевидно, что $q_2(x_0) = q_1(x_0) = y_0$ и $q_2(x_1) = q_1(x_1) = y_1$ для любых значений величины c_2.
Найдём её, исходя из условия $q_2(x_2) = y_2$.
Имеем $q_2(x_2) = q_1(x_2) + c_2 \cdot (x_2 - x_0) \cdot (x_2 - x_1) = y_2$. Отсюда

$$c_2 = \frac{y_2 - q_1(x_2)}{(x_2 - x_0) \cdot (x_2 - x_1)}$$

В принципе, дальнейшее уже более или менее понятно, но давайте распишем ещё один шаг процесса.

Ищем q_3 в виде

$$q_3(x) = q_2(x) + c_3 \cdot (x - x_0) \cdot (x - x_1) \cdot (x - x_2)$$

Множитель c_3 находим, исходя из условия $q_3(x_3) = y_3$. Условия $q_3(x_0) = y_0$, $q_3(x_1) = y_1$ и $q_3(x_2) = y_2$ очевидно выполняются вне зависимости от выбора c_3.
Имеем $q_3(x_3) = q_2(x_3) + c_3 \cdot (x_3 - x_0) \cdot (x_3 - x_1) \cdot (x_3 - x_2) = y_3$. Отсюда

$$c_3 = \frac{y_3 - q_2(x_3)}{(x_3 - x_0) \cdot (x_3 - x_1) \cdot (x_3 - x_2)}$$

Ну, думаю, что теперь уже всё ясно полностью, так что напишем общий вид перехода от q_{k-1} к q_k.

$$q_k(x) = q_{k-1}(x) + c_k \cdot (x - x_0) \cdot (x - x_1) \cdot (x - x_2) \cdot \ldots \cdot (x - x_{k-1}),$$

где

$$c_k = \frac{y_k - q_{k-1}(x_k)}{(x_k - x_0) \cdot (x_k - x_1) \cdot (x_k - x_2) \cdot \ldots \cdot (x_k - x_{k-1})}$$

Повторяем процесс вплоть до k=N.

Числовой пример. Построение интерполяционного многочлена Ньютона.

Повторю входные данные: $x_0=-2$, $y_0=1$, $x_1=1$, $y_1=-4$, $x_2=2$, $y_2=5$, $x_3=4$, $y_3=3$.
Начинаем. $q_0(x) = 1$.

Строим $q_1(x) = q_0(x) + c_1 \cdot (x-x_0) = 1 + c_1 \cdot (x+2)$.
$c_1=(y_1 - q_0(1))/(x_1 - x_0)$;
$q_0(x_1) = q_0(1) = 1$;
$c_1 =(-4-1)/1-(-2)) = -5/3$.
Получили $q_1(x) = 1 - 5/3 \cdot (x+2)$.

Строим $q_2(x) = q_1(x) + c_2 \cdot (x- x_0) \cdot (x- x_1) = 1 - 5/3 \cdot (x+2) + c_2 \cdot (x+2) \cdot (x-1)$.

$c_2=(y_2 - q_1(x_2))/((x_2-x_0)\cdot(x_2-x_1))$;
$q_1(x_2) = q_1(2) = 1 -5/3\cdot(2+2) = -17/3.$
$c_2=(5 – (-17/3))/((2-(-2))\cdot(2-1)) = 8/3;$
Получили $q_2(x) = 1 - 5/3\cdot(x+2) + 8/3\cdot(x+2)\cdot(x-1).$

И, наконец, строим $q_3(x) = q_2(x) + c_3\cdot(x- x_0)\cdot(x- x_1)\cdot(x- x_2) =$
$= 1 - 5/3\cdot(x+2) + 8/3\cdot(x+2)\cdot(x-1) + c_3\cdot(x+2)\cdot(x-1)\cdot(x-2).$
$c_3=(y_3 - q_2(x_3))/((x_3-x_0)\cdot(x_3-x_1)\cdot(x_3-x_2))$;
$q_2(x_3) = q_2(4) = 1 - 5/3\cdot(4+2) + 8/3\cdot(4+2)\cdot(4-1) = 39.$
$c_3=(3 – 39)/((4-(-2))\cdot(4-1)\cdot(4-2)) = -1;$
Получили $q_3(x) = 1 - 5/3\cdot(x+2) + 8/3\cdot(x+2)\cdot(x-1) –(x+2)\cdot(x-1)\cdot(x-2).$

Окончательно, мы получили выражение интерполяционного многочлена в форме Ньютона

$$p(x) = 1 - 5/3\cdot(x+2) + 8/3\cdot(x+2)\cdot(x-1) –(x+2)\cdot(x-1)\cdot(x-2).$$

Как и в случае с интерполяционным многочленом в форме Лагранжа, раскроем скобки, чтобы убедиться, что мы опять получили тот же самый многочлен, только иначе записанный.
$p(x) = 1 - 5/3\cdot(x+2) + 8/3\cdot(x+2)\cdot(x-1) –(x+2)\cdot(x-1)\cdot(x-2)$
$= 1 - 5/3\cdot(x+2) + 8/3\cdot(x^2+x-2) –(x^3-x^2-4x+4)$
$= -x^3 + (8/3+1)\cdot x^2 + (-5/3+8/3+4)\cdot x + (1-10/3–16/3-4)$
$= -x^3 + 11/3\cdot x^2 + 5\cdot x - 35/3$

Да. Это именно он!

В разделе задач к занятию приводятся примеры применения интерполяционого многочлена для прилизительного вычисления некоторых величин. А сейчас, в завершение занятия, я приведу потрясающее по красоте применение интерполяционного многочлена – конструкция, чудо которой само по себе окупает все усилия, затраченные на понимание сегодняшнего сюжета.

Схема Шамира разделения секрета.

Представим себе такую ситуацию. Есть группа людей, караулящих “красную кнопку”. Если поступает приказ, они должны нажать её, дав тем самым команду на запуск ракет. Служба несложная, но психологически очень тяжёлая. Любой из дежурных может сбрендить, нажать кнопку безо вского приказа и начать ядерную войну. Чтобы обезопасить мир, надо сделать так, чтобы кнопку можно было нажать только если вся группа соберётся вместе и все готовы нажать кнопку. Сделать это легко. Можно снять с каждого отпечатки пальцев и кнопка будет срабатывать, только если все они нажали её одновременно. А можно, например, поместить кнопку в бункер, навесить на дверь 10 замков или сколько там есть дежурных, у каждого дежурного есть ключ от своего замка, и, понятное дело, чтобы открыть дверь в сейф, они должны собраться все вместе и каждый должен открыть свой замок. Ладно, я придумал совсем уж чёрную сказку, давайте скажем, что нам надо подсчитать голоса на каком-нибудь очень важном голосовании. И, чтобы никто из членов избирательной комиссии не мог сфальсифицировать результат, каждый голосующий

помещает свой выбор в ящик, закрытый на сколько надо замков. И открыть ящик могут только все члены избирательной комиссии, собравшись вместе.

Однако у этих решений есть недостаток. Какой-нибудь хранитель "красной кнопки" может сбрендить в другую сторону и уничтожить свой ключ, сделав доступ невозможным. Любой один из членов избиратеной комиссии может сорвать подсчёт голосов таким же образом.

Давайте потребуем, чтобы для открывания замка требовалось согласие не всех людей, а только определённого их числа. Иначе говоря, чтобы никакие K из N носителей ключей не могли открыть замок, а любая группа из K+1 человека могла это сделать. Что-то вроде голосования. Если взять K=N/2, то это принятие решения большинстовм голосов, а если брать K = ⅔N или ¾N (бывает по-разному), то это уже называется квалифицированным большинством.

Вот эта задача и называется задачей разделения секрета: разделить секрет межде N людьми. В данном случае разделить так, чтобы никакие K из N людей не могли его узнать, а любые K+1 из них, совместно, могли весь этот секрет узнать.

Схема Шамира как раз и решает эту задачу, в цифровом виде, конечно. Идея метода состоит в том, что многочлен степени K полностью определяется любыми своими (K+1)-ой точками, но не меньшим их количеством.

Опишем реализацию метода.
Генерирование ключей. Выбираем случайный многочлен $p(x) = a_0+a_1x+a_2x^2 +…+a_Nx^N$. Случайно (или еще из каких-то соображений) выбираем все коэффициенты многочлена, кроме a_0. А вот a_0 – это как раз и есть главный секрет, магическое число, которое открывает замок. Выбираем N произвольных различных чисел x_1, x_2, … x_N. Ограничение здесь единственное – ни одно из этих чисел не должно быть равно 0. Понятно почему: дело в том, что p(0) – это и есть тот самый наш большой секрет, ведь $p(0)=a_0$. Других ограничений на выбор этих чисел нет, в том числе и к их какой-то там особой случайности, можно вовсе взять их равными 1, 2, 3, ..., N. Каждый из хранителей секрета получат свою долю – каждый из них получает какой-то x_i из этого набора и $p(x_i)$.
Восстановление секрета. Собираются K+1 носитель своей доли секрета, объединяют свои знания и строят интерполяционный многочлен. Многочлен этот, как мы знаем, построить можно (и мы знаем, как это сделать), при этом такой многочлен существует в единственном экземпляре, т.е.мы получим именно тот самый многочлен p. Коэффициент a_0 этого многочлена – это и есть большой секрет.
Невозможность восстановить секрет по K долям. Обратная история. Собираются K носителей секрета. Объединяют свои знания. Не можем пока восстановить. И тут приходит к ним змей-искуситель и говорит: "Вот вам ещё одна долька – моя пара чисел (0, S)". "Ура! – отвечают ему K носителей. - Давайте-ка быстро-быстро восстановим многочлен p по имеющимся у нас (K+1)-й точке и узнаем секрет". И что они получат? Они получат S, ведь свободный член (коэффициент a_0 многочлена так называется, если вдруг кто не знает, хотя это вряд ли) многочлена как раз и равен p(0). Излишне говорить, что змей-искуситель подсунул ребятам крутку, так что получить они могут с равными шансами любое число.
Что ещё хорошего в схеме Шамира.

1. Мы можем по ходу пьесы увеличивать количество носителей доли секрета. Не факт, что это так уж хорошо, но может оказаться полезным.
2. Мы можем изменять доли секрета, не изменяя сам секрет. Скажем, мы хотим каждый месяц изменять доли секрета, ну, типа за месяц они уже окажутся под большой угрозой быть скомпрометированными. Изменяем многочлен p, не изменяя его свободного члена, и раздаём новые значения многочлена p (можно даже не изменять иксы) носителям.
3. Можно раздавать неравные доли секрета. Ну, скажем нажать "красную кнопку" могут только три маршала из пяти, которые владеют долей секрета. Но ещё мы хотим, чтобы президент мог единолично нажать "красную кнопку". Ну так дадим ему три доли секрета – три пары (x, p(x)).

Задачи и упражнения.

1. Ну, конечно же, написать программу, которая строит интерполяционный многочлен. Тут даже три задачи – применить (уже написанную на прошлом занятии, написанную ведь, правда же?) процедуру решения систем линейных уравнений; построить интерполяционый многочлен в форме Лагранжа (и дополнительная задача – привести его к стандартному виду); построить интерполяционый многочлен в форме Ньютона (и тоже дополнительно – привести его к стандартному виду),
2. Несколько упражнений на приблизительное вычисление каких-нибудь интересных величин – интересующих нас значений каких-то функций.

2.1. Вычислить $\sqrt{105}$, проинтерполировав функцию $\sqrt{x}$ по её точкам
   a. (81, 9) , (100, 10), (121, 11)
   b. (10^2, 10), (10.2^2, 10.2), (10.5^2, 10.5)
   c. (10^2, 10.0), (10.1^2, 10.1), (10.2^2, 10.2), (10.3^2, 10.3), (10.4^2, 10.4), (10.5^2, 10.5), (10.6^2, 10.6)
   d. (9.5^2, 9.5), (9.6^2, 9.6), (9.7^2, 9.7), (9.8^2, 9.8), (9.9^2, 9.9), (10^2, 10.0), (10.1^2, 10.1), (10.2^2, 10.2), (10.3^2, 10.3), (10.4^2, 10.4), (10.5^2, 10.5)

2.2. Аналогичным образом вычислить приближённо $\sqrt[3]{30}$.

2.3. Вычислить sin 20^o. В качестве точек интерполяции брать точки с абсциссами 0^o, 15^o , 30^o, 45^o, 60^o, 75^o, 90^o .
   Для справки:

$$\sin 15^\circ = \frac{\sqrt{3}-1}{2\sqrt{2}}, \quad \sin 75^\circ = \frac{\sqrt{3}+1}{2\sqrt{2}}$$

3. Пример задачи на оценку качества интерполяции. Построить интерполяционный многочлен p(x), совпадающий с функцией cos(x) в нескольких точках.Например, в точках (0, 1), (pi/6, sqrt(3)/2), (pi/4, sqrt(2)/2), (pi/3, ½), (pi/2, 0).
   Задача состоит в том, чтобы найти все экстремумы функции f(x) = p(x) – cos(x) на отрезке [0; pi/2] и, соответственно, найти наибольшее отклонение интерполяционного многочлена на этом отрезке.

   Конечно, этот же приём можно применить и к любой другой функции и на любом отрезке.

4. Ну, и схему Шамира можно реализовать, хотя там вопрос больше пользовательского интерфейсса.